Elasticsearch：

Elasticsearch是一个实时分布式搜索和分析引擎。它让你以前所未有的速度处理大数据称为可能。

它用于全文搜索、结构化搜索、分析以及将这三者混合使用。

Lucene：不是全文检索引擎，是全文检索引擎的架构，提供了完整的查询引擎和索引引擎。是一个简单的工具包，方便在目标系统中实现全文检索的功能了。

Elasticsearch特点：

1. 分布式的实时文件存储，每个字段都被引擎索引并可被搜索。
2. 分布式的 实时分析搜素引擎
3. 可以扩展到上百台服务器，处理pb级结构化或非结构化数据。

Elasticsearch为java用户提供的两种客户端：

- 节点客户端：节点客户端以无数据节点身份加入集群，换言之，它自己不存储任何数据，但是它只读数据在集群的位置，并且能够直接转发请求到对应节点上。
- 传输客户端：轻量级的传输客户端，能够发送请求的到远程集群。它自己不加入集群。知识简单转发请求给集群中的节点。

传输协议为Elasticsearch Transport Protocol。

Elasticsearch的工作原理

1. 在Elasticsearch中，文档归属于一种类型 type，而这些类型存在索引index中。
2. 可以包含多个索引indices，每个搜索引擎可以包含多个类型types，每一个类型包含多个文档documents行，然后每个文档包含多个字段fields 列

Elasticsearch索引的数据结构

倒排索引：又叫反向索引，通过单词索引全文的索引。便于搜索的数据结构。

正向索引：当用户发起查询，搜索引擎会扫描引库中二点所有文档，找出所有包含文件此的文档。这样一次从文档中查询是否有关键词的方法。通过全文查找单词的方法。

单词文档矩阵

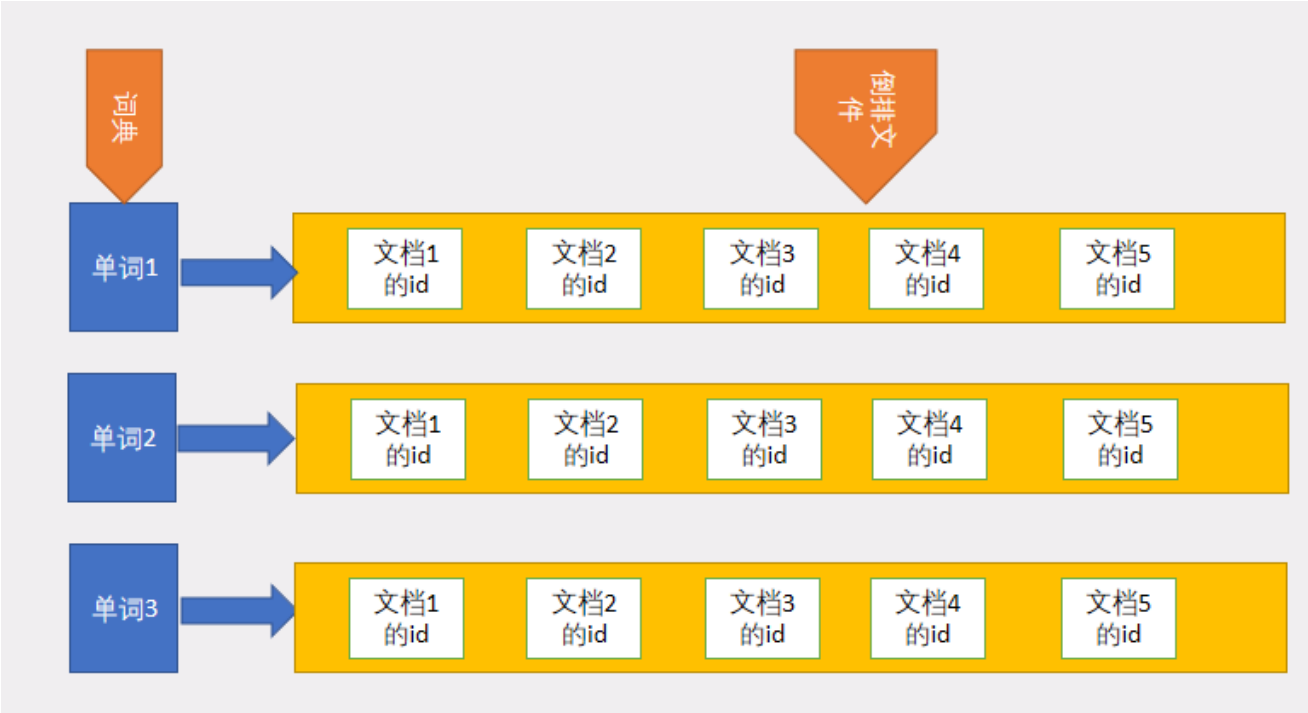
单词-文档矩阵是表达两者之间所具有的一种包含关系的概念模型。

- d1:乔布斯去了中国
- d2:苹果今年仍能占据大多数触摸屏产能
- d3:苹果公司首席执行官史蒂夫乔布斯宣布，ipad2将上市
- d4:乔布斯推动了世界，iphone、ipad、ipad2，一款一款接连不断
- d5:乔布斯吃了一个苹果

文档矩阵：

	d1	d2	d3	d4	d5
苹果		√	√		√
乔布斯	√		√	√	√
ipad2			√	√	

文档矩阵就是倒排索引的一种存储方式。可以通过倒排索引，获取包含这个档次的文档列表。倒排索引主要由两个部分组成：“单词词典”：“倒排文件”



TF（term frequency）：单词在文档中出现的次数。

pos：单词在文档中出现的位置

单词 id（wordid）	单词（word）	排列表（docid；TF；<Pos>）
1	乔布斯	(1;1;<1>);(3;1<6>)
2	…	…
3	..	..

pos:单词出现的位置

tf：单词出现的频率

wordid：单词id

es的增删查改

put方法增加

```
PUT /megacorp/employee/1
{

"first_name" : "John",

"last_name" : "Smith",

"age" : 25,

"about" : "I love to go rock climbing",
```

```
"interests": [ "sports", "music" ]

}
```

get方法查询

```
GET /megacorp/employee/1
```

head查询是否存在

delete 删除

查询年龄大于30的员工

```
GET /megacorp/employee/_search


{ "query" : { "filtered" : { "filter" : { "range" : { "age" : { "gt" : 30 } <1> } },"query" : {
"match" : { "last_name" : "smith" <2> } } } } }
```

filtered过滤器。gt： granter than

es的分布式

es致力于隐藏分布式系统的复杂性。以下操作都是在底层自动完成的。

- 将文档分区到不同的容器或者分片中，他们可以存在于一个或多个节点中。
- 将分片均匀的分配到各个节点，对索引和搜索做负载均衡
- 冗余每一个分片，防止硬件故障造成的数据丢失。
- 将集群中任意一个节点上的请求路由相应数据所在的节点。
- 无论是增加节点还是移除节点，分片都可以做无缝的扩展和迁移。

集群健康

green 表示主要分片和复制分片都可用

yellow表示主要分片可用，但不是所有复制分片都可用

red不是所有的主要分片都可用

es分片

一个分派你是一个最小级别单元。一个分片是一个Lucene实例，并且它本身就是一个完整的搜索引擎。

分片是es在集群中分发数据的关键。分片想象成数据的容器。文档存在分片中，然后分片分配到你集群中的节点上。当你爹集群扩展或者缩小。es将会自动在节点间迁移分片。

es索引

指向一个或多个分片的逻辑命名空间。

文档元数据

_index文档存储的地方

索引类似于关系型数据库里的数据库——是我们存储和索引关联数据的地方

_type文档代表的对象的类

就是类，对象。每个类型都有自己的映射

_id文档的唯一标识

每个字符串标识一个id

创建索引文档

文档通过其 _index 、 _type 、 _id 唯一确定

zookeeper

cap原则： c（consistency）数据一致性，a（available）可用性，p（partition tolerance）服务容错性

eureka 满足AP 可用性和容错性。当网络故障是，eureka的自我保护机制不会立即剔除服务，虽然用户获取到的服务不一定是可用的，但至少能够获取服务列表，用户访问服务列表时，还可以利用重试机制，找到正确的服务。

zookeeper满足cp，数据一致性和容错性。在数据全部同步之后才会返回给用户。